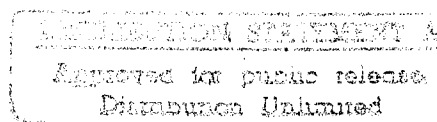PSR Report 2531

AIRBORNE OCEAN WATER LIDAR (OWL)
REAL TIME PROCESSOR (RTP)

M. Hryszko

March 1995

Phase Report

19950414 075

DTIC QUALITY INSPECTED 3

DISCLAIMER NOTICE

THIS DOCUMENT IS BEST QUALITY AVAILABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF COLOR PAGES WHICH DO NOT REPRODUCE LEGIBLY ON BLACK AND WHITE MICROFICHE.

REPORT DOCUMENTATION PAGE

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE March 1995	3. REPORT TYPE AND DATES COVERED Phase Report	

4. TITLE AND SUBTITLE

Airborne Ocean Water Lidar (OWL)
Real Time Processor (RTP)

5. FUNDING NUMBERS

C

N66269-90-C-0551

6. AUTHOR(S)

M. Hryszko

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Pacific-Sierra Research Corporation
2901 28th Street
Santa Monica, CA 90405

8. PERFORMING ORGANIZATION REPORT NUMBER

PSR Report 2531

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Naval Air Warfare Center Aircraft Division Warminster
Code 45560DR07
Warminster, PA 18974

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Distribution unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

The Hyperflo Real Time Processor (RTP) was developed by Pacific-Sierra Research Corporation as a part of the Naval Air Warfare Center's Ocean Water Lidar (OWL) system. The RTP was used for real time support of open ocean field tests at Barbers Point, Hawaii, in March 1993 (BARB I field test), and Jacksonville, Florida, in July 1994 (EMERALD I field test). This report describes the system configuration, and accomplishments associated with the preparation and execution of these exercises. This document is intended to supplement the overall test reports and provide insight into the development and use of the RTP. A secondary objective is to provide basic information on the capabilities, versatility and expandability of the Hyperflo RTP for possible future projects. It is assumed herein that the reader has knowledge of the OWL system, field test operations, general lidar processing methods, and basic computer architecture.

DTIC QUALITY INSPECTED 3

14. SUBJECT TERMS					15. NUMBER OF PAGES 48
lidar	ocean	laser	Hyperflo	real time	
digital signal processing		DSP			16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

ii

ABSTRACT

The Hyperflo Real Time Processor (RTP) was developed by Pacific-Sierra Research Corporation as a part of the Naval Air Warfare Center's Ocean Water Lidar (OWL) system. The RTP was used for real time support of open ocean field tests at Barbers Point, Hawaii, in March 1993 (BARB I field test), and Jacksonville, Florida, in July 1994 (EMERALD I field test). This report describes the system configuration, software development, and accomplishments associated with the preparation and execution of these exercises. This document is intended to supplement the overall test reports and provide insight into the development and use of the RTP. A secondary objective is to provide basic information on the capabilities, versatility and expandability of the Hyperflo RTP for possible future projects. It is assumed herein that the reader has knowledge of the OWL system, field test operations, general lidar processing methods, and basic computer architecture.

Accesion For		
NTIS CRA&I	☑	
DTIC TAB	☐	
Unannounced	☐	
Justification		
By		
Distribution /		
Availability Codes		
Dist	Avail and / or Special	
A-1		

TABLE OF CONTENTS

TABLE OF CONTENTS (continued)

LIST OF FIGURES

LIST OF TABLES

SECTION 1

INTRODUCTION

1.1   Overview

The Naval Air Warfare Center's (NAWC) Ocean Water Lidar (OWL) system operates at a peak pulse repetition frequency approaching 500 Hz. The system generates and records an immense amount of data for each lidar return. A High Density Digital Recorder (HDDR) is used to record the data stream. Up until 1991, real time feedback was limited to oscilloscopes, a 1-Hz waveform display and various hexadecimal LEDs at key locations around the OWL system. Proper operation was verified via a second or third generation copy of the data from HDDR data tape to 8-mm Exabyte tape and finally to a hard disk file. The turnaround time for ground debugs, engineering checkouts, or in-flight problems could take many hours or days. A need existed for a real time capability to investigate and analyze the data stream on a non-interfering basis.

The processing requirements are fairly stringent for the OWL system. Signal processing and all other computations must be completed within two milliseconds per sample to maintain real time operation without loss of data. A user interface and a graphical display are also required for practical employment. Since the OWL system is a research and development tool, future OWL system upgrades might require additional processing capabilities. Therefore, the computer hardware needs to be expandable, allowing for unforeseen processing requirements.

For a traditional single processor computer design, the data throughput and processing load would require a supercomputer. A custom analog real time processing design was attempted unsuccessfully in the past and was not considered for the current task. Recently developed inexpensive multiprocessor boards could be used, but most were not complete systems. In 1990, *Hyperflo*, designed by Pacific Cyber/Metrix Corporation, was chosen by NAWC because it was the only expandable, fully integrated general purpose multiprocessor computer specifically constructed for real time solutions.

Employing a data-flow architecture, the Hyperflo system provides multi-processor operation without need for explicit parallelism in the problem to be solved. The system can be custom-configured with an array of different microprocessors and VMEbus compatible boards in a single 19-inch card cage. As a computer system, Hyperflo provides a complete solution for real time multiprocessing. Hyperflo provides multiprocessor hardware, a real time data-flow operating system, and a complete set of software development tools.

1.2   Type 1 Intercept Real Time Processor

In 1991 Pacific-Sierra Research Corporation (PSR) was tasked to implement the company's Type 1 intercept algorithms and displays on the Hyperflo system. The program was originally conceived for use on an IBM compatible PC and had to be restructured to operate in real time. Under a team effort, NAWC supplied the integrated hardware, software routines for general purpose graphic functions, and the HDDR interface and PSR was responsible for the final application, using the individual pieces of the hardware and combining the software into a single program.

Concurrently, NAWC developed a Hyperflo-based system health application. The program was used to monitor key housekeeping parameters of the OWL system, ensuring that the system would stay within normal operating conditions. The application was used in conjunction with the Type 1 intercept software. Only the Type 1 intercept program or the system health application could operate at any given time.

PSR's software was evaluated in the October 1992 and the January 1993 engineering checkout tests, denoted as JAX V and JAX VI, respectively. Some improvements were recognized that would increase the utility of the software. The application was upgraded prior to the March 1993 field test, denoted as BARB I. During the test flight operations, over 120 Type 1 intercepts were observed. The real time feed back of the applications significantly enhanced system preparation and test conduct. The Hyperflo provided a unique capability to monitor, support, and analyze data for the OWL system.

1.3   Real Time Dye Map

PSR was tasked in October 1993 to develop a subsurface dye layer mapping application, because of the success and proven performance of the Type 1 intercept application. The software was more complex in scope, including an entire new operator interface, integration of four-channel linear processing, and additional hardware support. The application named *Hydro* was used during the EMERALD I field test in July 1994 at Jacksonville, Florida. The primary objective of the Hydro program was to provide a detailed GPS-positioned map of the subsurface dye. The data were collected, combined in the aircraft and transmitted as a map via a radio link to a surface research vessel. The map was used to make real time decisions concerning the conduct of the test. Halfway through the test, the dye layer was remapped, primarily because the dye was expected to drift. A second map was transmitted to the surface vessel, and evaluated to determine if changes should be made in conducting the test.

The Hydro application enabled a complete survey of the subsurface dye shape. This was previously unattainable and was a crucial element in the decision chain for test conduct. A dye map was successfully constructed during all dye test flights in which the aircraft participated. The application successfully provided accurate maps during all such dye tests and was critical to the success of the field test. The Hydro application was also used for the

Type 1 intercept portion of the exercise. The dye return signal is very similar to the Type 1 return but at a much higher level. The dye processing thresholds were lowered producing some limited success locating the Type 1 intercepts.

1.4   Future Tests

The Hyperflo system was designed to be expandable. Current applications have not fully exploited the capabilities of the present system configuration. If required, additional specialty processing boards could be added to increase computational power significantly. The net result is a base unit which could be expanded to meet practically any algorithm requirements. In general, without major system upgrades (such as increased laser pulse repetition frequency) or 3-D image processing tasks, the current Hyperflo hardware setup is adequate for most applications.

SECTION 2

HYPERFLO SYSTEM DESCRIPTION

2.1 Hyperflo Hardware Description

The Hyperflo RTP described in this report is a part of the OWL system. Two essentially identical Hyperflo units are used in support of operations. One unit is installed aboard P-3A aircraft BuNo 152150, stationed at NAWC Aircraft Division Warminster (NAWCADWAR). Figure 2.1 is a cutaway view of the aircraft showing the RTP location in relation to the other equipment. The second unit is used for data transcription and as a software development machine. This unit is located with the other ground support equipment. Parts of the two units are interchangeable and are used as spares.

The Hyperflo receives data from the HDDR on a read-after-write basis. This insulates the Hyperflo from the other components of the OWL system and allows for transparent operation in real time or playback modes. The Hyperflo can operate in an HDDR feed-through mode, allowing for normal RTP operation without recording the data on tape.

Two different Hyperflo hardware configurations were used to support the two field tests, and the two engineering checkout tests. Table 2.1 lists the Hyperflo items used for the tests. The configurations are essentially identical except for an additional DSP-2 board, serial board and mouse, and a PCMCIA board used in the EMERALD I test.

Table 2.1. Hyperflo Hardware Test Configuration

Item	JAX V / JAX VI / BARB I	EMERALD
SRB-2 Board	1	1
GMS-1 Board	1	1
MPU-3 Board	1	1
DSP-2 Board	1	2
Custom Interface Board	1	1
RAM Pack Board	1	1
Eltec E-5 OS-9 Board	1	1
3.5-inch Floppy Drive	1	1
Serial Interface Board with Mouse	0	1
Eltec OPAC Graphics Board	1	1
19-inch Color Monitor	1	1
PCMCIA Board	0	1
PC Laptop	1	1
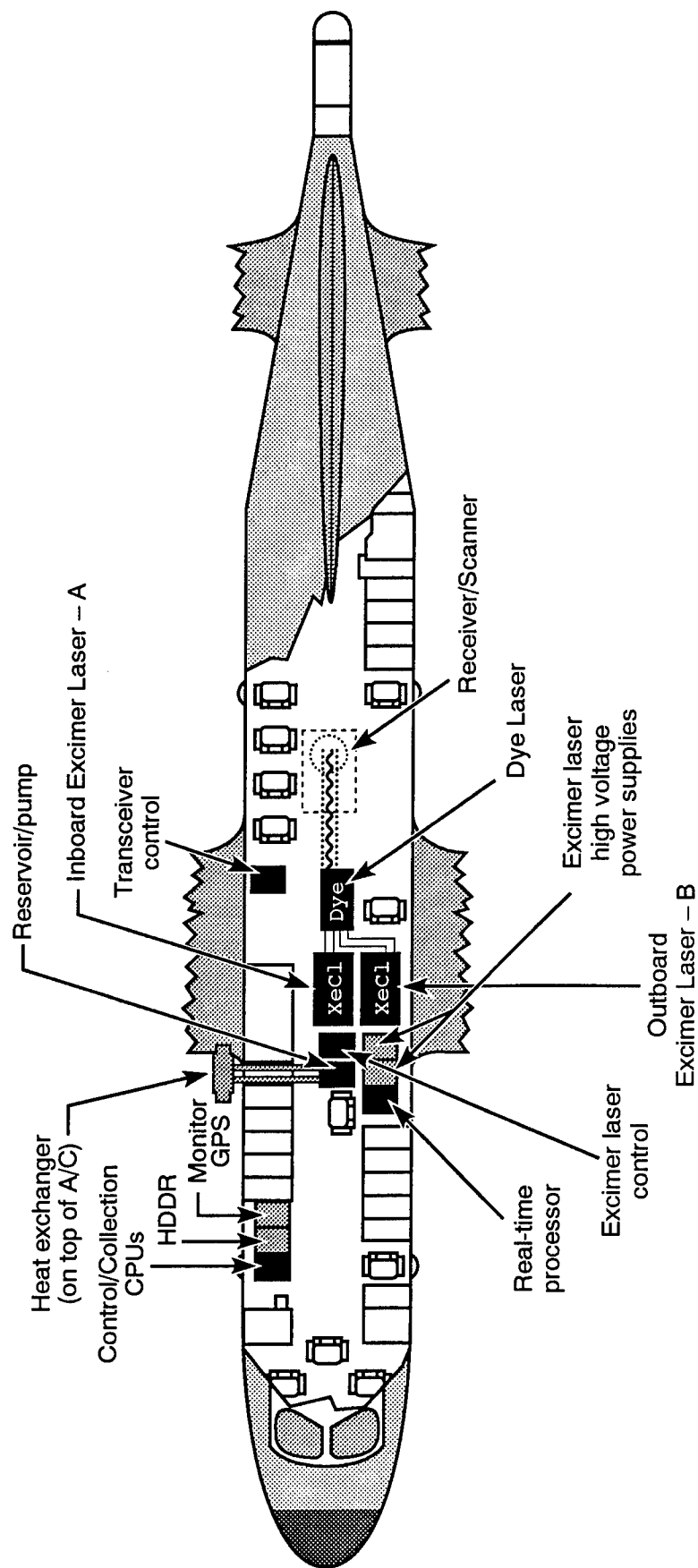
Figure 2.1 Layout of lidar equipment in the aircraft.

Reservoir/pump

Inboard Excimer Laser – A

Transceiver control

Receiver/Scanner

Dye Laser

Excimer laser high voltage power supplies

Heat exchanger (on top of A/C)

Control/Collection CPUs

HDDR

Monitor

GPS

Outboard Excimer Laser – B

Real-time processor

Excimer laser control

5

The boards are installed in a single 19-inch 21-slot VMEbus back plane heavy-gauge rack-mounted aluminum chassis, with a 1000-W power supply, and a high-performance, forced-air cooling system. The following subsections describe the components of the Hyperflo.

2.1.1 SRB-2 Board

The SRB-2 System Resource Board is a system controller board for the Hyperflo system. The SRB-2 provides VMEbus arbiter functions, programmable system clock, system control, program loading and fault detection. User and system interaction with the SRB-2 take place through a set of ASCII and coded high level commands. Most of these operations of the SRB-2 board are transparent and automatically exchanged between SRB-2 and the other Hyperflo boards in the system during normal operation. Other commands are directly available and serve system control and supervision functions. It is not required to write code or directly interact with the hardware on the SRB-2 board.

2.1.2 GMS-1 Board

The GMS-1 Board is a global-level data exchange unit that is designed to maximize the data transfer rates over the VMEbus. The board provides bulk memory for applications and for system administrative functions. A portion of this bulk memory is implemented as EPROM storage for the Hyperflo operating system (FLOS). (FLOS is a run-time only distributed operating system intended for code execution and debugging.) The GMS-1 Board is configured with 4 Mbytes of memory as installed in the OWL system, with 3.5 Mbytes usable for applications. The board is capable of responding to normal or extended addressing, although the Hyperflo system normally uses extended addressing.

2.1.3 MPU-3 Board

The MPU-3 is a general purpose multiprocessor board designed specifically for operation in the Hyperflo. Five Motorola 25-MHz 68030 processors and five 68882-floating-point coprocessors are capable of executing 28 million instructions per second (MIPS) with 10 million 80-bit floating-point operations per second (FLOPS) for user applications. The board provides 512 kbytes of RAM for each processor with four of the processors available for user applications and the fifth processor serving as the on-board master.

2.1.4 DSP-2 Board

The DSP-2 is a high performance, digital signal processor (DSP) board compatible with the Hyperflo system. The DSP-2 includes three Texas Instruments TMS320C30 DSP chips running at 33 MHz with a board capacity of up to 50 MIPS and 100 32-bit MFLOPS.

Each DSP device has 512 kbytes of private memory and is interconnected with high speed bidirectional first-in-first-out (FIFO) memory. Additionally, each DSP is connected via FIFO to a front panel connector. The FIFOs were originally configured as 512 words deep prior to October 1993, and afterwards expanded to 1024 words. The DSP-2 board includes a 68030/68882 processor pair which serves as the board master.

2.1.5 Custom Interface Board

A custom interface board that was designed and fabricated at NAWCADWAR is used to connect the HDDR to a front panel port on a DSP-2 board. The board takes the 28 signal lines of the HDDR and clocks the digital data into the DSP-2 FIFO. In March 1992, a system line and a housekeeping data line were swapped in the recorded format. With a front panel switch on the custom interface board, the two data lines can be changed to allow for transparent playback in both recording modes.

2.1.6 RAM Pack Board

A removable twin RAM pack board is installed in the Hyperflo system. Each RAM pack is a removable battery-backed-up module capable of storing 2 Mbytes of data. The total board capacity is 4 Mbytes. One of the RAM packs is used to hold essential system boot-up programs, the other is used to store user applications. The RAM packs are formatted as OS-9 disks and the packs can be manipulated with standard disk operations. Seven RAM packs were available for use. Four packs were taken onboard for normal mission operation: a boot pack, a backup boot pack, an application pack, and a backup application pack. The RAM packs are not normally used in the Hyperflo ground support unit other than to load the application software.

2.1.7 Eltec E-5 OS-9 Board

The Eltec E-5 board is installed in the Hyperflo system to provide an Operating System-9 (OS-9) interface with the RTP. (Hyperflo supports an OS-9 or UNIX interface.) The board configured with a 68030 processor enables system bootup, disk media access, Hyperflo tool execution, Hyperflo application loading and control, and terminal interface. The ground station Hyperflo unit's OS-9 board is used to develop 68000 series code and create the final Hyperflo applications.

2.1.8 3.5 inch Floppy Diskette Drive

The Hyperflo unit is equipped with a 3.5-inch, 690-kbyte floppy disk drive. The drive is used to boot up the OS-9 system with software contained on the bootup RAM pack. The

drive was used to copy RTP target detection logs from the RAM pack to a blank floppy diskette after each target detection test flight.

2.1.9   Serial Interface Board with Mouse

A serial interface board was installed in the Hyperflo to enable an RS-232 serial mouse connection. The mouse is connected via a custom interface cable fabricated by NAWCADWAR. A mouse driver application, which was provided by NAWCADWAR, distributed mouse movements and button status to other applications by request.

2.1.10   Eltec OPAC Graphics Board

A high performance Eltec OPAC graphics board was used to provide the Hyperflo with a color graphic capability. The OPAC board is equipped with two Advanced Micro Devices Quad Pixel Dataflow Manager Am95C60 chips. The OPAC supports up to 64 colors and four overlay colors from a palette of 16 million at a 1280- by 1024-pixel resolution. The OPAC can manage high speed screen formats up to 400 MHz. Three front panel connectors provide the RGB signals from the OPAC. The horizontal synchronization is contained within the green signal.

2.1.11   19-inch Color Monitor

A 19-inch 50-MHz 1280- by 1024-pixel resolution color monitor is connected via the RGB front panel of the OPAC board. This monitor is used to provide a variety of color displays for the Hyperflo system. The monitor contains a front panel switch for degaussing the screen and two knobs which adjust the color and contrast of the monitor.

2.1.12   PCMCIA Board

A PCMCIA board is used by the Hyperflo to access credit-card-size, removable-battery-backed-up memory cards. The board contains four slots for the insertion of the 2-Mbyte cards. The board was set up with a flat extended 32-bit addressing. The cards must be used in pairs, because memory access is performed with odd addresses on one card and even addresses on the other. Although the total board capacity is 8 Mbytes, two cards with a capacity of 4 Mbytes were used typically during flights. (Seven PCMCIA cards were available.) The cards can be write-protected, and since they use a standard PCMCIA interface they can be accessed by any PCMCIA capable device.

2.1.13 PC Laptop

An IBM PC compatible laptop was used with the Hyperflo system. The laptop connects to a front panel RS-232 serial port of the OS-9 board and provides, via Kermit communication software, an unsophisticated VT-100 terminal emulation. The laptop can be connected to the front panel of the SRB board, but typically this is not required. A higher-end PC laptop was used as a DSP development platform, but this function is not supported during RTP operation.

2.2 Hyperflo Architecture Description

The Hyperflo achieves superior computing performance through a multi-instruction-stream, multiple-data-stream design and the natural pipe lining of data flow architecture. The basic principle of data flow consists of passing information from one modular process to another. This structure closely parallels the approach used to solve a complex problem. Specifically, the flow charts and block diagrams that are employed to illustrate a complicated problem can be used as the basis for the application design. Real time realization consists of implementing the structure with sufficient microprocessors to handle the operation. Throughput can be increased simply by adding processors. Alternatively, old processor boards can be augmented or replaced by newer boards, or by special purpose processors. Thus, the integrated system is unlikely to be driven to obsolescence. With the advance in semi-conductor technology and/or with increased application resource demand, the Hyperflo can be upgraded rather than discarded.

There are a number of pathways which allow high speed data transfers to sustain the Hyperflo's high performance. Three primary methods were used in the BARB I and EMERALD I applications to transfer data from processor to processor: high speed FIFOs between the DSPs, data channels for all inter-processor data flow, and message links for inter-application communications. In general, the large data bandwidth requirements demand dedicated FIFOs for data transfers, which are available only with the DSP processors. Data channels are used for operations of less demanding bandwidth. Both processor types can use this pathway. The message links are used for non-time-critical, low-bandwidth operations between concurrently executing applications. This is available only with the 68030 processor.

2.2.1 DSP FIFOs

Each of the processors on the DSP board are hardwired together via dedicated high speed bidirectional FIFOs. The DSPs are also connected to the front panel, enabling external input of data or high speed transfers to other DSP boards. The size of the FIFOs was 512 words by 32-bits in length for the BARB I field test. They were expanded to 1024 words by 32-bits in length for the EMERALD I test. The FIFOs operate on a first-in-first-out basis coupled to high speed memory. FIFOs can be interrogated to determine a 3-level status as

full, half full, or empty. The FIFOs are constructed to prevent new data from clocking into the FIFO during a full condition.

Since the FIFOs are small, data exchanges larger than the FIFO buffer size require dedicated processor time. Both sending and receiving processors need to be on-line for the transfer; therefore the processors cannot continue other operations until the data exchange is completed. This limits the flexibility of the application design but, if properly programmed, it does not significantly affect the speed of the application. Additional throughput is achieved by using on-chip RAM, which is limited to 2048 long words. Normal 'C' code programming and optimization do not exploit this high speed, quick-access memory. The on-chip RAM was not used with the BARB I application but was fully investigated and used with the EMERALD I program.

The DSPs can be instructed to transfer data via an interrupt service routine (ISR) keyed on FIFO data availability. In practicality this method is unusable, because all processing must be completed before any new data arrive. Additionally, ISRs significantly tax the overall processing budget, because the ISR and program code must be swapped during processor operation. ISRs were investigated for the EMERALD I application, but performance degradation precluded their use.

The DSPs are equipped with direct memory access (DMA) controllers, which allow data transfers without interfering with the operation of the processor. The DMAs can be keyed to FIFO status level to provide data transfers in background. Typically, DMAs are used to interface with slow external memories or peripherals. On the Hyperflo, DMA transfers can move from 2 to 255 long words. This capability was insufficient for most uses in the BARB I and EMERALD I applications.

2.2.2 Data Channels

Data channels are the general data transfer mechanism for the Hyperflo system. A data channel consists of a pre-defined memory block connected to the data ports of the program modules. Channel size can vary from 2 kbytes to the limits of the available system resources. Each channel works on a first-in-first-out basis similar to the DSP FIFOs, but the channels are not hardwired. Modules on different processors, the same processor, or different boards can be connected with a channel. Memory allocation and connection are made dynamically by the operating system during application loading.

Channels are linked to modules through ports. Each module can have up to eight read ports and eight write ports. To allow data exchanges between different processor types, a signed or unsigned long integer format is used. Status level checks permit the interrogation of the ports connected to the channels. The returned value depends upon the type of port which is examined. Input ports return the number of bytes which can be read from the channel. Output ports return the number of bytes which can be written to the channel.

10

2.2.3   Message Links

The message passing support routines allow for the exchange of text messages between the calling module and any other agent in the Hyperflo system. These services are most typically used in the interaction between a user at a terminal or console, and the module. The message link is also used to provide mouse movement information to other modules. Once the link is established between concurrently running applications, messages can be retrieved from the module's report buffer. The software support routines in the Hyperflo 68030 'C' libraries do not support multiple open links from the same source.

2.3   Hyperflo Software Description

The Hyperflo system is supplied with a number of software tools helpful in creating an application which runs under the FLOS operating system. The Hyperflo software was available in two development environments, OS-9 and UNIX. The OS-9 version was selected because of prior NAWCADWAR experience in OS-9 development and the substantial cost saving versus a UNIX system. (The OS-9 software is no longer sold or supported by Pacific Cyber/Metrix.)

2.3.1   Application Network Configurator (ANC)

ANC is a utility that is used to connect modules (.mod) into an application network and generate an application network map (.anm) and module block files (.mbl) which are necessary to run the program on Hyperflo FLOS. Modules are loaded into the utility and the ports of the modules are connected to define the data channels of the application. Through ANC, the channel specifications are defined and particular application requirements such as module processor pre-assignments are outlined. Application information is stored in a file (.net) by ANC and can be reloaded to modify the network. If changes are made to a module's frame, the module must be removed from the network, reloaded, and reconnected to the channels.

2.3.2   Data Channel Monitoring (DMON)

DMON is a Hyperflo application which is used to observe data flow in channels. Once DMON is loaded, the user selects the range of channel numbers to be monitored. DMON links with the user's VT100 compatible terminal and provides periodic updates on the status of the selected channels. The utility is helpful as a debugging tool, by supplying information to determine data flow problems or data jams. DMON displays channel status information such as bytes in the channel, average data level as a percent of capacity, percentage of the time that the channel is empty and full, and the minimum and maximum data level as a percentage of capacity.

2.3.3   Module Frame Editor (MFE) and Frame Generation Utility (FGU)

The MFE is used to define the module's operating characteristics (basic functional description, read and write port specifications, etc.). This module frame (.def) is converted into an assembly language relocatable (.r) file for 68030 target modules with the FGU program. For the DSP processors, the frame definition file is modified with the FGUC30 utility to create a C30 assembly source. The FGU outputs are linked with the compiled processor source code to form a module (.mod) format file required by the ANC program.

2.3.4   Remote Terminal Program

The Remote Terminal program is an OS-9 utility for communicating with the Hyperflo system. The Remote Terminal program implements a subset of the commands available through the system console attached to the SRB board. With the Remote Terminal program, applications can be loaded or terminated and Hyperflo system status can be viewed. The Remote Terminal program provides a capability to link to running applications and issuing keyboard commands. The Remote Terminal program displays any system exceptions which may have occurred during operation or program loading.

2.4   Application Design

Hyperflo application design requires matching the bandwidth needs of a particular data transfer with the processor's capability. Real time specifications might direct the application to process 2048 waveform bytes and 384 bytes of ancillary data at a rate of 500 samples per second. This is a total input of over 1.2 Mbytes per second, just to receive the data. Afterwards, the processor performs some calculations and sends all relevant data, which may include some partially processed data, to the next processor. About 2 milliseconds per sample is required to perform all processor functions, including all signal processing and data flow.

Since data flow is critical to real time performance, the method of implementing data dependent program execution is crucial. The data pathways can be examined for data availability, or space for transmitting data. Program operation can be dependent upon this status. Thus, a subroutine could be keyed on availability of data from a pathway. If the data are unavailable, then other subroutines could be performed or other pathways could be checked.

2.5   Hyperflo Development Environment Description

A Hyperflo application consists of a number of modules linked together and operating as a unit to accomplish a task. Two types of processors are available in the Hyperflo for a module, a Motorola 68030 and a Texas Instruments (TI) TMS320C30. Module development

takes place separately in two different operating environments for the two processors. An OS-9 based system with a 68020 series optimizer, 'C' compiler, assembler and linker is used for 68030 target code. An MS-DOS system with a TI TMS320C30 optimizing 'C' compiler, assembler and linker is used for the C30 target code.

2.5.1 68030 Development

The 68030 processor software development is fairly straightforward. All aspects of development can be done in the OS-9 operating environment. After the module characteristics are decided, the module frame is constructed via the MFE program. The program code is written and a makefile is used to compile and link all the different 68030 application modules and create separate .mod files. The 68030 modules were written in 'C' without need for any hand-coded assembler routines. The programming structure for the 68030 compiler was very similar to standard ANSI 'C'. Only a subset of the standard 'C' functions was available for use on the 68030s with the Hyperflo system.

2.5.2 C30 Development

C30 module development is more complicated. First, the module frame is constructed in the OS-9 operating environment. The frame definition file (.def) is modified with the FGUC30 program to generate a C30 assembly frame source code. The source is transferred to an MS-DOS operating environment via a cross platform communications program. Then, the C30 module code and the frame source code are compiled and linked with the TMS320C30 optimizing compiler, assembler and linker using a makefile. The completed module is transferred back to the OS-9 system for input into the ANC program.

There are a couple of programming considerations for C30 development. All integral types (char, short, int, long, and their unsigned counterparts) are equivalent and are represented as 32-bit binary values. All floating point types (float, double, and long double) are equivalent and are represented in TMS320C30 32-bit floating point format. Since all data types are stored in 32-bit representations, memory address locations are accessed as 32-bit words using 24-bit pointer range. This yields 16 Mwords of addressable memory space. Note, sub-word or byte level addressing is not available.

Significant program performance gains are obtained by using the 2 kwords of high-speed on-chip RAM located from address 0x809800 to 0x809FFF. The internal RAM can be accessed in 'C' by creating a pointer to the RAM block. Normal 'C' programming and compiler optimization do not access this area. The on-chip RAM was used only in the assembly routines in the BARB I applications. For the EMERALD I application, the internal RAM was exploited for all C30 routines.

13

2.5.3 Application Development

After all modules are programmed, the '.mod' files generated for the specific 68030 or C30 processors are located in a single OS-9 directory and inputted into the ANC program. The read and write ports of the modules are interconnected, specified and formed into data channels. The modules can be targeted for a particular processor at application start time or allowed to load into any compatible processor. C30 modules should be targeted to a specific processor location if FIFOs are used, because these connections are hardwired together.

SECTION 3

BARB I APPLICATION

3.1   BARB I Application Overview

An application was created from January 1992 to March 1993 to support the 14-24 March 1993 field test at Barking Sands, Hawaii. The data set designation for the exercise and the Hyperflo application name in this report are BARB I, although many different names were used during application development and test. BARB I was not a single unique program, but a series of modifications and improvements that culminated with an application used at the field test.

Two engineering checkout exercises were conducted at Jacksonville prior to the BARB I field test. The exercises, noted as JAX V and JAX VI, were held during 21-25 October 1992 and 7-14 January 1993, respectively. The tests were used to evaluate the functionality and operability of the software. Improvements and suggestions generated from the evaluation were incorporated into the application up until the commencement of the BARB I test. Only one minor application adjustment was required after the start of the field test. Afterwards, the application did not have any further problems.

The BARB I application was very successful, logging over 120 real time Type 1 intercepts during the course of the test. Type 1 optical depth observations ranged from 2.0 to 4.0 Kd (diffuse attenuation coefficient and depth product) at 480 nm and 2.0 to 5.0 Kd at 532 nm. Post test analysis revealed the same optical depth processing limits as that observed in real time. Therefore, the real time Type 1 intercept algorithm performed as well as the post-flight non-real time methods.

The BARB I test was the first in-flight use of the Hyperflo hardware and associated software under field test conditions. Data collection efficiency during the exercise was significantly increased because of the ability to determine OWL system health and test objectives in real time. The Hyperflo proved to be a valuable, reliable and a much-needed tool for the lidar. The on-board computer system provided a situational awareness which significantly enhanced the ability of the project crew to collect high quality data.

3.2   BARB I Application Background

The BARB I application is based on PSR's Type 1 intercept algorithm. This processing method was originally implemented as an IBM PC compatible DOS application and was selected by the project community for conversion into the Hyperflo environment. Figure 3.2 shows a typical IBM PC Type 1 display of a one meter diameter white sphere deployed at a depth of about 45 m, corresponding to 2 Kd. The data set (N0618) was collected during the January 1993, JAX VI engineering checkout test. This display format

15

provides the means to visualize the intercept in showing simultaneous overhead and depth profile views of the data. (Figure 3.2 is described in detail in Section 3.2.2.)

The PSR Type 1 intercept algorithm was a strong candidate for porting to the Hyperflo system, because it consistently provided data realizations down to an optical depth of 4.5 Kd in a single post-processing pass. This continuous execution is critical for real time operation, because intercept confirmation is required immediately to affect the test conduct.

3.2.1 PSR's Type 1 Intercept Algorithm

A number of steps were used to process OWL waveform returns for a Type 1 intercept. The overall algorithm was ported to the Hyperflo, although the majority of the PC code had to be altered. The general processing method is as follows:

a. de-emphasize waveform
b. surface align on 3 dB down of peak return
c. power normalize waveform near surface
d. determine processing bin range
e. subtract background on a bin-by-bin basis
f. update background statistics, bin-by-bin
g. filter waveform, bin-by-bin
h. modify filtered output by variance, bin-by-bin
i. update variance statistics, bin-by-bin
j. determine peak output bin
k. modify output based on previous peak output statistics
l. update peak output statistics.

The processor output is statistically driven, self-adjusting to the variation in the environmental background. All the statistics are coupled to filters and require a start-up period for optimal operation. The number of waveforms required for initialization can be modified by changing the filter coefficients of the statistics. For the BARB I application, the constants were selected so that it took about 1000 waveforms from the outset of the processing for the filters to settle. To decrease this time, the first waveform was used as the mean background. If that waveform was a typical background return, the start-up sequence was reduced.

The lidar return statistics are calculated on a bin-by-bin basis for the background and variance waveforms. This approach is used to determine the peak anomalous bin in the processed interval. The output is compared statistically to previous outputs and categorized into five output types as follows:

a. level 1, output $\leq 1\ \sigma$ from mean output
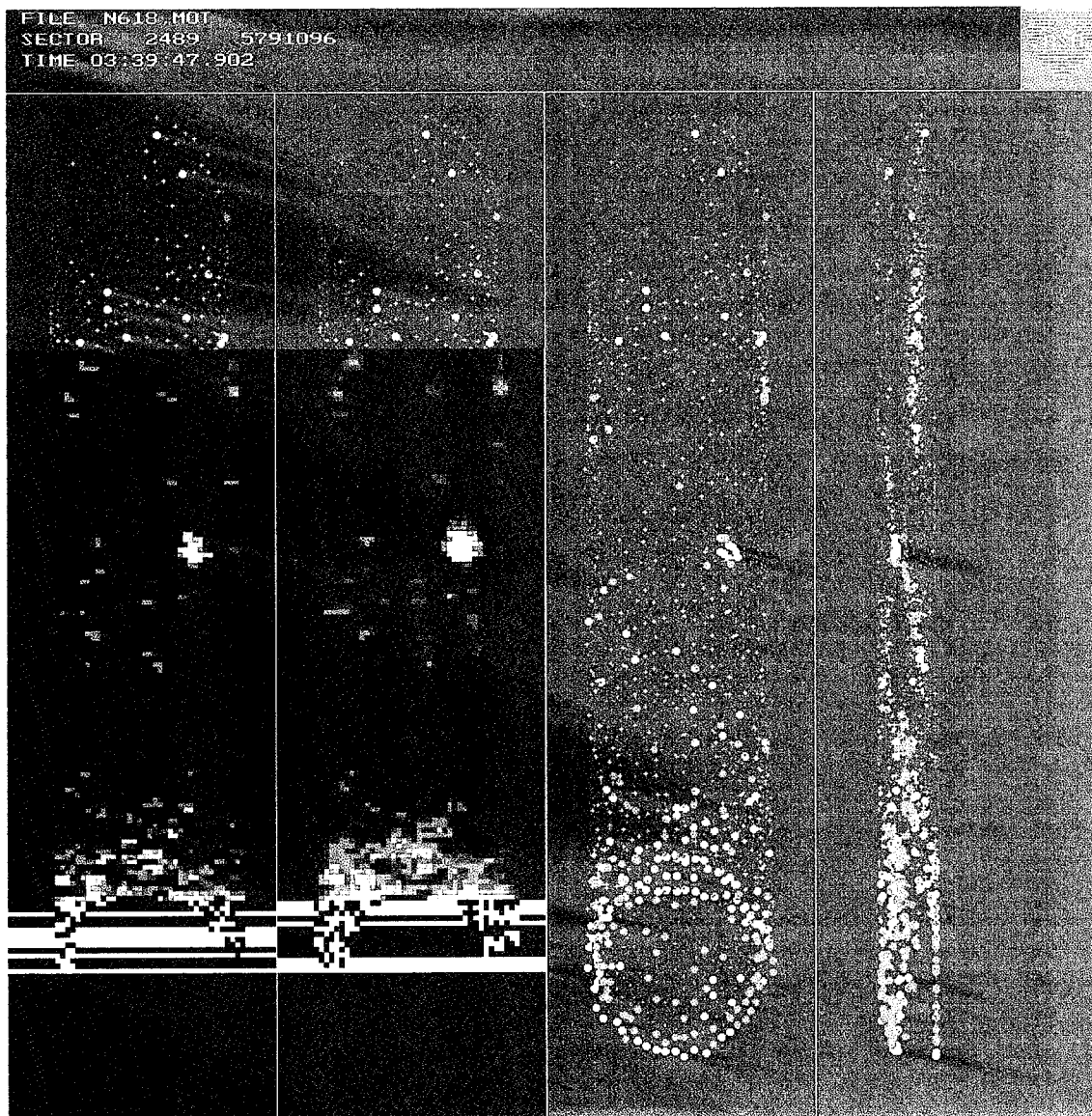b. level 2, $1\ \sigma <$ output $\leq 2\ \sigma$ from the mean

Figure 3.2 Typical IBM PC intercept display of a 1-m white sphere
at a depth of around 2 Kd (45m).

17

This page left blank intentionally

18

     c.   level 3, $2\,\sigma <$ output $\leq 3\,\sigma$ from the mean

     d.   level 4, $3\,\sigma <$ output $\leq 4\,\sigma$ from the mean

     e.   level 5, $> 4\,\sigma$ from the mean.

The advantage of the statistical processing approach is that it can be performed in a continuous single pass through the data. The output is self-adjusting for environmental variations, but extremely strong anomalies can cause the filters to under-compensate the subsequent waveforms. Under most circumstances, this is not a problem, because these anomalies usually occur from the intercept returns. Alternatively, significant waveform errors (corrupted waveforms) could produce the same effect. Considerable care is required to identify and reject these waveforms, to prevent an unwanted contamination of the processing statistics.

3.2.2   PSR's Type 1 Intercept Display

The five-level output from the PSR's Type 1 intercept algorithm is used to create displays that provide a simultaneous overhead and depth profile view through the data. In figure 3.2, the third panel from the left displays the aircraft overhead view of the processor outputs. The fourth panel shows an along-track depth aspect of the same data. The processor outputs in the overhead view are positioned in aircraft track relative position compensated for roll, pitch and drift. The depth view shows the output depths from 0 to 150 m from the left to the right side of the panel. The five processor output levels are represented in the display as follows: (Non-surveyed areas are in a blue background color.)

     a.   level 1, single light blue pixel

     b.   level 2, single red or orange pixel

     c.   level 3, 2 pixel radius red or orange circle

     d.   level 4, 3 pixel radius red or orange circle

     e.   level 5, 3 pixel radius white circle.

Output levels 2 through 4 orange and red colored circles represent front and back scan positions respectively. The scan color separation permits a coherence evaluation of the data to help distinguish intercept signatures from false alarms. A white circle represents a very significant anomaly which most likely occurred from an intercept. The front or back scan information is not distinguished for the level 5 white circle.

The two left hand panels of figure 3.2 show 2-D filtered representations of the data in an overhead view. The raw output of the waveform processor is mapped into a rotating array and the data are input into a smoothing filter. The left panel presents the output at this stage. The second panel shows the result after the application of another filter, matched for the predicted output size of a one-meter sphere convolved with the laser spot. Both displays remove high frequency components of the mapped data and show a cleaned-up view. The colors in the 2-D displays, which progress from black through blue, green, yellow, red,

19

magenta and white, depict the increasing statistical significance of the mapped area compared to previous samples. The display has the same start-up filtering artifact as the other displays.

This 2-D filter display was not implemented into the Hyperflo BARB I application because of funding constraints. In principle, the algorithm conversion would have been fairly straightforward although screen real estate would have to be rearranged to accommodate the additional panel. Furthermore, the addition of the 2-D algorithm would have required at least one DSP-2 board to handle the extra calculations. The 2-D display does not necessarily increase the data realization capability for a well-trained observer, but it allows for the ability to quantitatively determine the coherence of the Type 1 intercept. This is the first step in producing an automated application.

3.3 BARB I Application Implementation

The BARB I application was created from the ground up on the Hyperflo system. The computer and associated processing boards were new to the market and relatively little programming experience was accessible. The OWL configuration is unique; programming functions for the application such as graphics drivers, console interface, HDDR link, etc. had to be developed in-house. Although the entire processing method had already been developed and implemented on an IBM PC, very little of the source code was usable when ported to the Hyperflo. The code had to be modified for processor considerations and for performance optimization.

3.3.1 BARB I Application Display

Figure 3.3.1 is a block diagram of the BARB I application's color display. The two long right hand blocks show an overhead and depth profile view of the data. The same five output levels of the PSR algorithm were used to reveal the strength and location of the waveform return. The second panel from the right (the overhead view) shows the aircraft cross-track versus along-track distance around the data. The far right panel (the depth profile view) shows a depth versus along track distance view of the data. Newer data samples overwrite previous samples when the data occur in the same screen areas. The current aircraft position stays fixed at the top of the screen and the previous samples are scrolled downward away from the aircraft.

3.3.2 BARB I Application Control and Console Interface

The BARB I application operator's console was used to input program start up values, determine processing regions, and reset program statistics. The console was also used to request a freeze copy of the bottom half of the overhead view and depth profile displays. The console interfaces to a keyboard and it displays text through a VT100 compatible terminal.
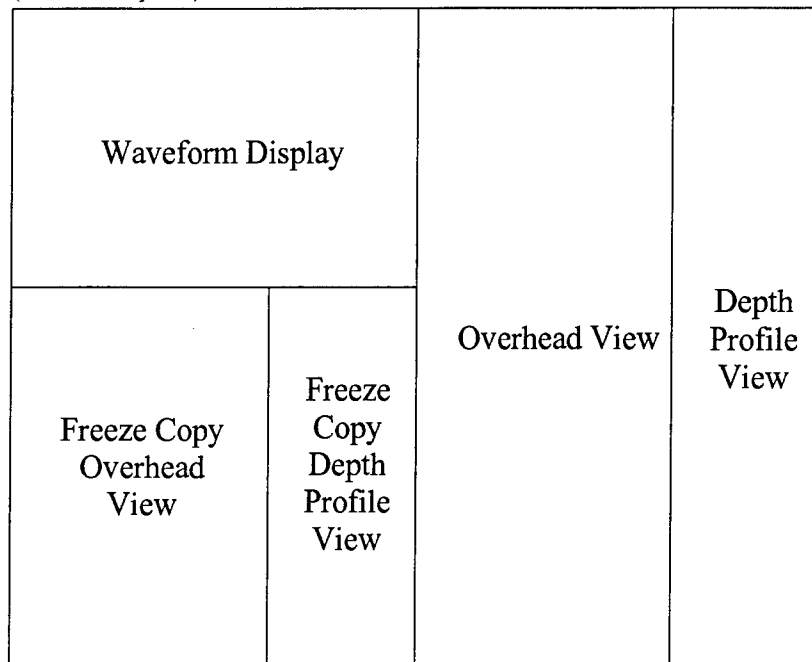
(1280 x 1024 pixels)

Waveform Display		Overhead View	Depth Profile View
Freeze Copy Overhead View	Freeze Copy Depth Profile View		

Figure 3.3.1 Block diagram of the BARBI application screen.

21

The console is connected to a 68030-type processor in the BARB I application via a message link and all communication with the program is handled through this interface.

The BARB I application requires a few important input parameters for proper program operation. These values are based upon the particular test configuration such as aircraft altitude, processing depth range, and receiver mode. Upon application start-up, certain default values are loaded. If waveform processing is initiated, these values will be used, otherwise the operator can modify certain parameters.

In order to maintain real time operation, programming compromises were needed which restricted the processing regions for the surface and search regions within the waveform. This did not reduce the detection capability of the application, but it increased the responsibility of the operator to ensure proper program configuration. Typically in the course of a test the operator would have four general concerns:

      a.   proper receiver mode operation (log or pre-emphasized log)
      b.   correct surface search region
      c.   correct depth search region
      d.   recording and logging of intercepts.

Proper receiver mode operation required matching the operating mode with the processing mode. Mismatch would cause degradation of the waveform signal from improper filter application. Normally one operating mode was used consistently during a flight and typically the log pre-emphasized receiver mode was preferred during the course of the test.

Setting the correct surface and processing regions is usually done during the first occurrence in a series of runs. An adequate surface region is selected which would encompass a surface which wanders with aircraft roll and pitch. Careful observation of the waveform is required in a two-level gain mode to select the correct surface region. In this mode, a second, higher gain is used just past the surface which boosts the deep signal return. The surface return no longer represents the peak output in the waveform, but a stationary glitch.

Correct selection of the processing region depends upon proper detection of the surface. The center of the area should be selected to coincide with the intercept depth, typically a total range of about 50 meters is selected. The processing range can be reduced to decrease the clutter from non-target regions. Selecting a small range runs the risk of not capturing the complete intercept signal.

The last operator responsibility is locating Type 1 intercepts and recording of BARB I application parameter inputs. This helps post-flight debug in the event of a BARB I application problem. A log file automatically records the program parameters whenever the operator uses the freeze copy option. The operator is required to record on the log sheet the visual strength of the signature, which helps identify data sets for post-flight analysis.

3.3.3 BARB I Application Log File

To assist post-test analysis, an ASCII text log file was stored on the RAM pack which recorded parameters whenever the user freeze-copied the scrolling overhead view and depth profile displays. This would be done during possible Type 1 intercept or unusual anomaly encounters. The file contained information such as data processing input parameters and calculated aircraft encounter times. Using a separate paper copy log sheet, the BARB I application operator visually interpreted and recorded the apparent strength of the realizations. Table 3.3.3 is a performance summary of the number of logged Type 1 intercepts at two operating wavelengths and various optical depths.

Table 3.3.3. BARB I Application Performance Summary

Kd	# Logged ($\lambda = 480$ nm)	
2.0 (~ 67 m)	41	
3.0 (~ 100 m)	31	
3.5 (~ 117 m)	24	
4.0 (~ 133 m)	15	Total = 111

Kd	# Logged $\left(\lambda = 532\text{nm}\left[K_{\lambda_{532}} = 2.0 * K_{\lambda_{480}}\right]\right)$	
2.0 (~ 33 m)	1	
3.0 (~ 50 m)	3	
4.0 (~ 67 m)	5	
5.0 (~ 83 m)	3	Total = 12

The BARB I application showed a large majority of the Type 1 encounters down to an optical depth of 4.0 and 5.0 Kd or 140 and 85 m for 480 and 532 nm, respectively. Additional intercepts were not seen or logged due to the time sharing with the system health application or possible application operator error. Post flight analysis of the data with the BARB I application was not performed to determine the probability of success, nor was an evaluation started to determine the probability of false alarm. The post-test analysis did reveal the BARB I application performed as well as non-real time processing methods. That is, the post test methods did not realize an optically deeper Type 1 intercept than was identified and logged in real time.

SECTION 4

EMERALD I APPLICATION

4.1   EMERALD I Application Overview

Based on the success of the BARB I application, PSR was tasked in October 1993 to develop the algorithms and displays for a GPS-positioned sub-surface dye mapping application on the Hyperflo computer. A real time aircraft-based dye mapping application was required because of the unpredictable sheet drift that occurred between the time of dye deployment and the start of the test. Since the OWL system was the primary sensor in the test, it was critical to initially map the dye shape with the lidar and transmit the airborne dye map view to the test conductor, who was located on a surface positioned research vessel designated as R/V 1. Without this capability, collection of high quality data would be in jeopardy because of the position accuracy of the other research vessels in relation to a strong dye mass.

Work on the EMERALD I application was started in January 1994. Many new basic features had to be developed which were not used in the BARB I application or were new to the OWL system such as a graphical interface, an upgraded data stream and a new GPS receiver format. Another major problem was the unavailability of OWL system dye data in the EMERALD I format. Greensheet I and II data were available but the data formats were significantly different and lacked consistent GPS position information.

Late in the application development, a four-channel linear receiver mode was integrated as an option for the OWL system. Reconstruction of the four channel linear data into a log waveform allowed a single algorithm to be used for most operating receiver modes (four-channel linear, log/linear split, and first channel log). The application was constructed to distinguish between these modes from the ancillary data and pre-process the waveforms accordingly. For dye mapping, a single channel log mode was used to ensure that the waveform was maintained inside the digitizer window. For the bulk of the test conduct, the four-channel linear receiver mode was the primary operating method.

Although, the EMERALD I application successfully accomplished all dye mapping requirements during the course of the test, minor processing problems were encountered. The modified thresholds under-predicted the strong dye area during the engineering checkout test and first flight and over-predicted the strong dye area during the second flight. This did not have a significant impact on test operations, because the gross dye mass was detected and mapped. A number of threshold iterations were needed prior to the final level which was agreed upon by the on-site project analysts. For the four remaining dye maps, the threshold levels were unchanged. Appendix A contains a sample dye intensity map for each of the seven dye operation days.

24

The dye maps were transmitted to the R/V 1 through a radio activated modem. In order to ensure timely transfers, the nearly 1-Mbyte map file was sub-sampled and compressed down to a file size of about 5 kbytes. One map was transmitted prior to each set of data runs; two sets of data runs were usually performed during the course of the day's operation. The maps were decompressed on the R/V 1 and reviewed to determine proper test configuration.

4.2   Dye Detection Algorithm

The dye detection algorithm developed for the EMERALD I test used the waveform returns collected during the Greensheet II test as the baseline. Generally the following method was used to detect dye during the EMERALD I test on a waveform by waveform basis:

a.   pre-process four-channel linear mode into log mode
b.   find location of first waveform peak (surface is not always peak return)
c.   surface align on 3 dB down point of first peak
d.   select 20 to 85 m as waveform processing region
e.   perform a least squares parabolic fit for the processing region (background waveform)
f.   subtract background to produce residual waveform
g.   band pass filter residual waveform for dye signature
h.   threshold peak output for dye/no-dye
i.   provide one byte output (0-no sample, 1-no dye, and 2-255 increasing dye intensity).

In addition to the output intensity, the processor provided dye depth and an estimated thickness. This information was transferred to the mapping algorithm portion of the application.

During the test, the dye detection algorithm had difficulty processing waveforms which were obstructed owing to clouds. The processor output indicated that there was no dye in the sample. The algorithm processed the waveform properly according to the program code, but since cloud-obstructed waveform rejection was not part of the processor, an incorrect evaluation occurred. The proper output should have been a non-valid sample taken at the location with a '0' processor output. Instead, a value of '1' was recorded for that location and any previous sample taken in the location was overwritten.

4.3   Graphical Interface and Display

The essentials of a mouse driven graphical interface was designed and written for the EMERALD I application on the Hyperflo computer. This significantly enhanced the user friendliness of the real time processor. Most processing selections and program operations were implemented as point and click functions on the 19-inch monitor. Only numerical

keyboard entries were required on the console. This allowed the user to maintain head-up operation of the software.

The mouse was connected to the Hyperflo through an RS-232 connector on a serial VME board. A custom interface cable was constructed between the mouse and the connector. A software application was written to provide mouse movement information to other application through a message link. The mouse location was displayed in the EMERALD I application with a cross hair that appeared across the screen. The mouse integration was essential for smooth operation of the graphical interface.

A large number of functions were written for the EMERALD I application to facilitate the graphical requirements on the Hyperflo. Multiple graphic planes were used to separate the aircraft cursor, map grid lines, text displays, mouse cross hairs, etc. from the map display. Special functions were written to optimize zooming, provide push buttons and display the optical angle sensor (OAS) data. A single dedicated 68030 processor was used to handle the graphical interface to the OPAC board and all graphic functions were pipelined through this module.

4.4   GPS Map Data

All research vessels participating in the EMERALD I test used a GPS referenced coordinate system. The map transmitted to the test conductor was required to be accurate within the commercial code GPS receiver specification. Verification of the EMERALD I application mapping accuracy was performed during a NAWCADWAR local flight at Lake Nockamixon, Pennsylvania. Figure 4.4 shows a surface return intensity map for a collection of data runs at Lake Nockamixon.

The intensity of a return from the water is significantly less than that expected from a land return. The green intensity of the sampled location is proportional to the likelihood of a land return. The shore outline can be observed as the boundary between the dark green/black and bright green areas. A south pier located in the west part of Lake Nockamixon was surveyed by a Magellan GPS Receiver. This location was compared to the position observed from the Hyperflo system and to an interpolated point from a United States Geological Survey (USGS) map. Table 4.4 shows the results of the comparison.

Table 4.4.   GPS Location Comparison of South Pier at Lake Nockamixon

Source	Latitude	Longitude
USGS map	40° 27.37 N	75° 14.13 W
Magellan GPS	40° 27.376 N	75° 14.136 W
Hyperflo	40° 27.37 N	75° 14.14 W

(20 x 20 km)
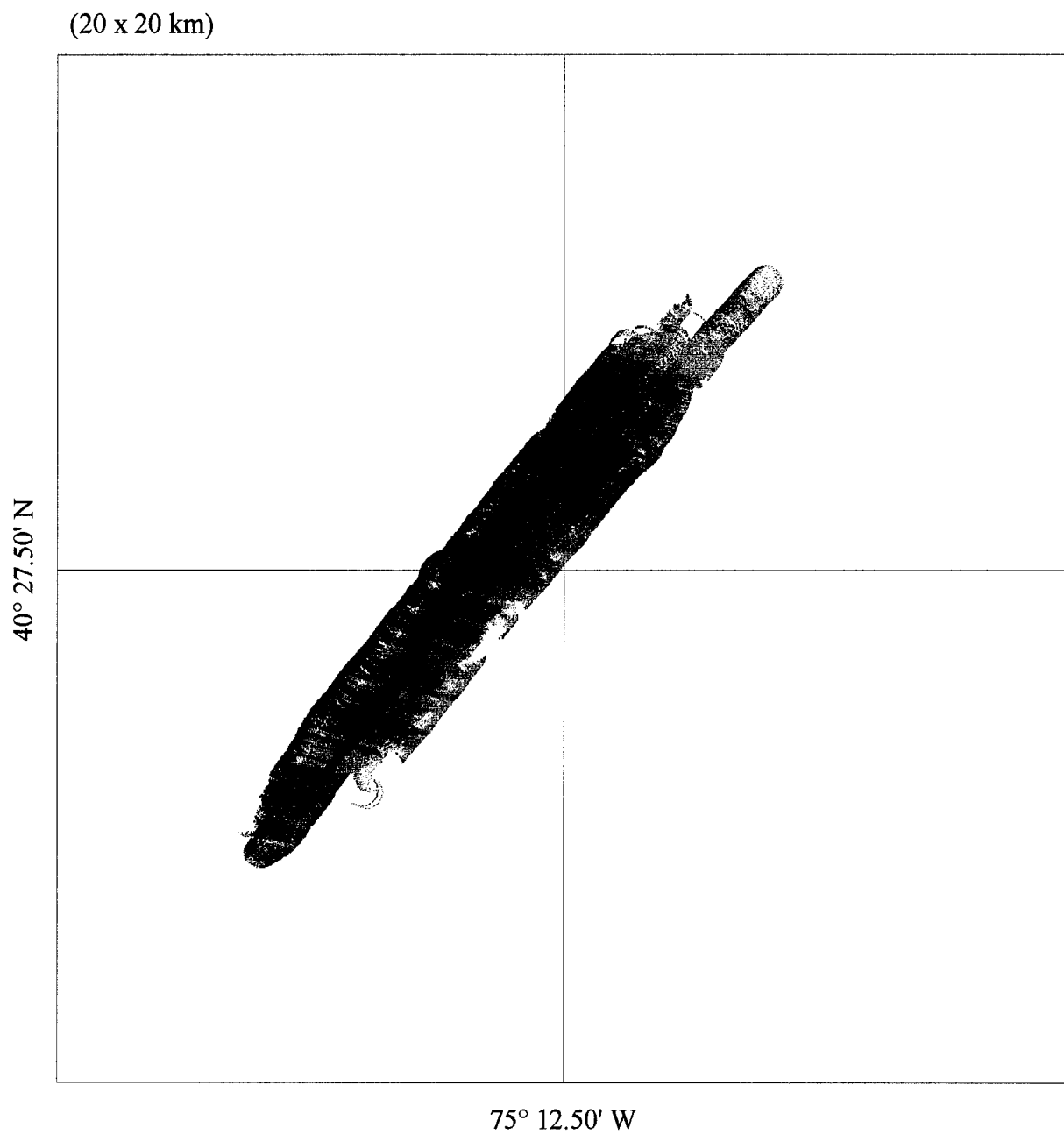

40° 27.50' N

75° 12.50' W

Figure 4.4 Surface return intensity map of Lake Nockamixon.

This page left blank intentionally

The Hyperflo pier position was obtained using the EMERALD I application by recording the site of a pier-like object projecting into the water. The red circle on figure 4.4 shows the location of the pier. All three pier position sources agreed to within about 20 m, well within the GPS commercial code specification. The USGS survey map was reduced to the scale of the Hyperflo map and a gross comparison was performed between the maps. The Hyperflo map matched well to the size and shape of the USGS map.

4.4.1 GPS Mapping Algorithm

To accurately map the points of incidence of the lidar pulses on the water surface accurately, a laser spot position adjusting algorithm was developed. GPS positions, provided in the OWL data stream at a 1-Hz rate, arrive with a variable latency which can be calculated by comparing the time of solution to the laser shot time. An assumption is made that the aircraft inertial velocity north and south components are very accurate on a short term basis (a few seconds). These inertial measurements are used to calculate the projected GPS position when a GPS update is received. Between GPS updates, the aircraft position is adjusted on a sample-by-sample basis using the same inertial measurements and the differential sample times. The calculations are performed on a DSP processor with 32-bit floating point numbers. Aircraft position changes of less than 20 cm are typical between lidar samples. To avoid severe rounding errors from adding relatively small numbers to large numbers, the GPS position was broken down into two components, a gross value with an accuracy of about a meter, and a sub-meter value. The GPS position was still subject to the intentional errors associated with the commercial receiver, but additional errors due to computations were nominally avoided.

4.5 EMERALD I Display

The 19-inch color display was separated into a number of areas each with a different operational function. Figure 4.5 is a block diagram of the EMERALD I display as implemented in the application. Two map areas were used to show the aircraft position and map information at two scales. The rest of the screen was used to show critical operational parameters, a waveform display and an OAS display. Most application operations were done through this color screen. Only numerical inputs were required through the keyboard console.

An aircraft concentric circle object and 10-km precursor are displayed on the maps whenever the aircraft position or precursor are within the scales of the operational or zoom maps. A track line can be drawn with the mouse on either map with the corresponding starting and ending latitudes and longitudes which are to be displayed. The exact coordinates of an intended aircraft track line can be inputted through a keyboard at the specific latitudes and longitudes within the aircraft operating area. During the exercise, the track lines were used to monitor the actual aircraft track versus the intended path over the dye area.
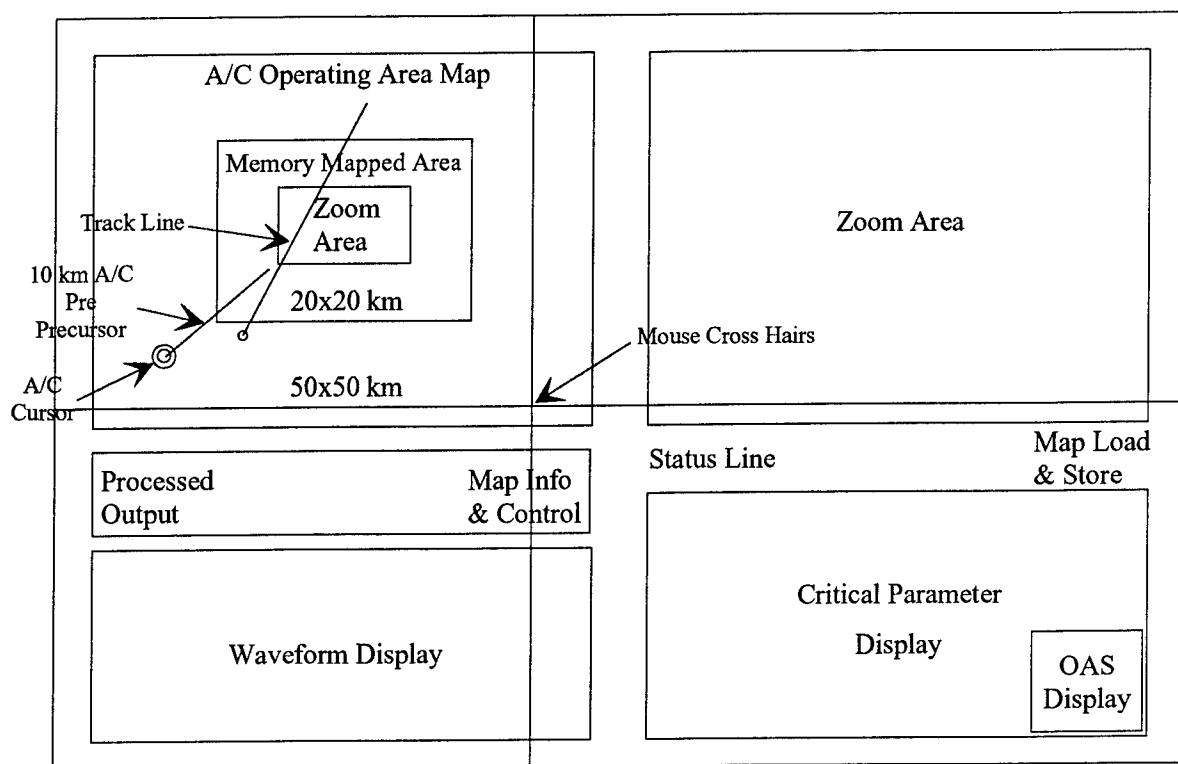
29

Figure 4.5 Block diagram of Emerald I display.

4.5.1 Aircraft Operating Area Display

The upper left hand section of the screen represents a 50- by 50-km aircraft operating area. The center location of the map is user selected to be nearly any point on the earth. A Mercator projection was used to transform the GPS latitude and longitude into screen positions. The map latitude center point was restricted to an absolute value less than 80° to prevent problems with the projection algorithm. (A Mercator projection contains singularities at the poles.) The excluded area was not within the normal operating region for the EMERALD I application. Located within the aircraft operating area is a 20- by 20-km memory mapped area. This region is mirrored in global memory with a 1000- by 1000-byte array corresponding to a 20- by 20 m square per element. Any sample output which is located within the target area is recorded in global memory. New samples replace older colocated samples. Within the target area is a variable sized square zoom selection area. The box displays the zoom size which would be used if a zoom selection is made. Only five zoom scales are allowed: 20 by 20, 15 by 15, 10 by 10, 5 by 5, and 2 by 2 km. The zoom area must be located completely within the target area.

Whenever the mouse is located within the limits of the aircraft operating area map, the mouse position is displayed in the map information and control area of the display. The position is displayed in degrees and decimal minutes in GPS relative coordinates. An intended track line can be drawn on the aircraft operating area map by pressing and holding the mouse button when the mouse cross hairs are located at the track line start point. Releasing the mouse button fixes the track line end point. If the mouse cross hairs wander past the border of the aircraft operating area map, the track line end point will occur near the border point. The starting point of the track line is designated with a small circle.

4.5.2 Zoom Area Display

The upper right hand location of the screen shows the current zoom selection area. The default zoom selection is a 20- by 20-km view upon program start-up. After the zoom function is selected with the map control buttons, the global memory locations corresponding to the map are mirrored and displayed on the screen. All subsequent laser samples which occur within the zoom area appear on the screen. The aircraft cursor and pre-cursor will appear on the zoom area map if these symbols fall within the selected limits. Only the portions of the track line which fall within the limits of the zoom area will appear on the zoom map. A track line which is drawn in the zoom area will appear only in the zoom map. The track line will not appear in the aircraft operating area.

4.5.3 Waveform Display

On the lower left hand portion of the application screen is a waveform display area. The application shows sample lidar return waveforms from the system at about a 1-Hz rate. The block has two different display methods, one for the four-channel linear operating mode

and another for all other modes. If four channels are selected from the digitizer configuration ancillary data, the raw inverted waveform is presented as a light gray trace and a reconstructed logged waveform is shown as a white line. For all other modes the raw inverted waveform is displayed as a white line.

Various additional lines present the processed output for the 4-channel reconstructed waveform or the first channel of all other modes. (The first channel is processed and assumed to be log for all other modes.) Dye processing output parameters such as one byte dye intensity value, dye depth, and thickness are shown in the processed output display area. On the log or logged waveform trace, vertical lines are positioned at the surface and dye locations. Overlaid on the waveform is a line corresponding to the parabolic fit to the logged exponential decay of the background.

4.5.4 Critical Parameter Display

On the lower right hand portion of the screen is the critical parameter display area. In this block are presented various important ancillary, calculated or application oriented parameters updated at about a 1-Hz rate. The block contains data on aircraft GPS position, INS, sample time, scan rate, DSP processor load, digitizer configuration, high resolution trigger information, OAS parameters, and surface return distance and statistics. The GPS and OAS data were color coded to indicate valid status. The GPS data is displayed in green if a recent valid GPS block is received. If a valid GPS block was not received in three seconds, the block would turn yellow. If a valid GPS block is not received within five seconds, the block turns and stays red. For the OAS parameters, the x-position and/or y-position turn red if the values exceed an absolute value of about 30,000 counts. Otherwise, the OAS parameters remain green displaying the most recent values.

4.5.5 OAS Display

Located in the critical parameter display block, below the OAS parameters, is the OAS graphical display and control area. Two push button controls vary the OAS display for two settings, 5 and 20 mrad. This controls the maximum size of the outer ring of the OAS's bull's-eye display. In the 5- and 20-mrad modes, five and four concentric rings correspond to 1- and 5-mrad increments, respectively. The OAS display shows the calculated location of all lidar waveforms. The screen is cleared and updated with new shot locations at about a 1-Hz rate. This gives the system operator feedback for proper operation of the OAS, ensuring that the laser beam remains centrally located in the bull's-eye. If drift is noted or a misalignment occurs, the operator can alert the project coordinator and a decision can be made for an in-flight beam realignment.

32

4.5.6 Map Load and Store

The map load and store area contains a number of push buttons which control map functions. The initial map center coordinates can be modified by selecting the 'LAT/LON' button. The operator inputs the center map location and confirms the coordinates by selecting the 'CONFIRM' button. The 'CLEAR' button allows the operator to clear the display and maintain the center coordinates of the map. Both these procedures reset the maps global memory and clear the aircraft operating area and zoom area displays.

The 'SAVE' button stores the current map selection by mirroring map global memory into two PCMCIA battery-backed-up RAM cards. Related map information is recorded in a header block. This includes GPS time that the map was stored, scale, latitude and longitude coordinates and the map resolution. The total map size with the header block is 1,001,000 bytes. The procedure requires several seconds to perform. the operation. (During the save function, most lidar samples are not properly processed.) Previously stored map data can be loaded from the PCMCIA cards by selecting the 'LOAD' button. A confirmation is required that the load function was requested. The PCMCIA map data overwrite the current data stored in global memory. The center coordinates of the map are not modified with the selection of the load function. The aircraft operating and zoom areas are updated to reflect the changes in the new map.

4.6 Dye Map Data Link

Upon completion of a dye map, the data were transmitted to R/V 1 to determine proper test conduct. The PCMCIA cards were removed from the Hyperflo and inserted into an IBM PC compatible with a PCMCIA card reader. An IBM DOS program was written to automatically read the map data, sub-sample and compress the map, and view the map on the screen prior to transmitting the map to R/V 1. The 1,001,000-original-byte file was compressed down to a final size of about 5,000 bytes. This high compression was achieved at a cost to resolution and single pixel scale. The 1000- by 1000-element array was sub-sampled to a scale of 500- by 500-elements with 2 bits of information per element. These 62,500 bytes of data were further compressed using a standard commercial compression program PKZIP 2.0. The data were transmitted to R/V 1 with a radio-keyed 1200-baud modem with a transfer time of about one minute.

33

REFERENCES

(a) *Airborne Lidar System Description and Calibration, Revision 4.1,* Pacific-Sierra Research Corporation and Naval Air Warfare Center Aircraft Division Warminster, PSR Report 2386, September 1993.

(b) *DSP-2 Technical Specification Sheet,* Pacific Cyber/Metrix, Inc., undated.

(c) *EMERALD Airborne Lidar System Description and Calibration,* Pacific-Sierra Research Corporation and Naval Air Warfare Center Aircraft Division Warminster, PSR Report 2469, Draft, May 1994.

(d) *Hyperflo Reference Manual, Revision 1.6,* Pacific Cyber/Metrix, Inc., 1991.

(e) *Hyperflo Product Description,* Pacific Cyber/Metrix, Inc., undated.

(f) *MPU-3 Technical Specification Sheet,* Pacific Cyber/Metrix, Inc., undated.

(g) *TMS320C30 Optimizing C Compiler Reference Guide, Revision D,* Texas Instruments Incorporated, Microprocessor Development Systems, SPRU034D, August 1990.

(h) *TMS320C3x User's Guide, Revision E,* Texas Instruments Incorporated, Digital Signal Processing Products, SPU031B, June 1991.

APPENDIX A

EMERALD I DYE MAPS

A1.1 EMERALD I Real Time Dye Maps

On the following pages are copies of the maps which were generated in real time during the EMERALD I pre-test exercise and field test in July 1994. The original map data contain a 256-level, 1000- by 1000-pixel array depicting a GPS-positioned subsurface dye layer. At each pixel location a single byte value represents the following information:

       a.   0 - no valid sample taken at location
       b.   1 - valid sample taken below dye threshold
       c.   2 through 255 - valid sample taken above threshold.

Of the seven maps included, one is taken from a pre-test exercise and six are representative maps for each of six days. The following seven files were taken from a catalog of 64 maps stored during the course of the exercise:

       a.   19405_12.INT
       b.   19903_38.INT
       c.   20006_24.INT
       d.   20301_58.INT
       e.   20402_00.INT
       f.   20801_55.INT
       g.   20905_22.INT.

The file names indicate the day and time in 1994 that the map data was recorded. The first three digits represent the Julian day of the year followed by the Greenwich mean hour and minute. All 64 raw data map files were distributed to the project community.

Intensity Map 19405_12.INT

(20 x 20 km)
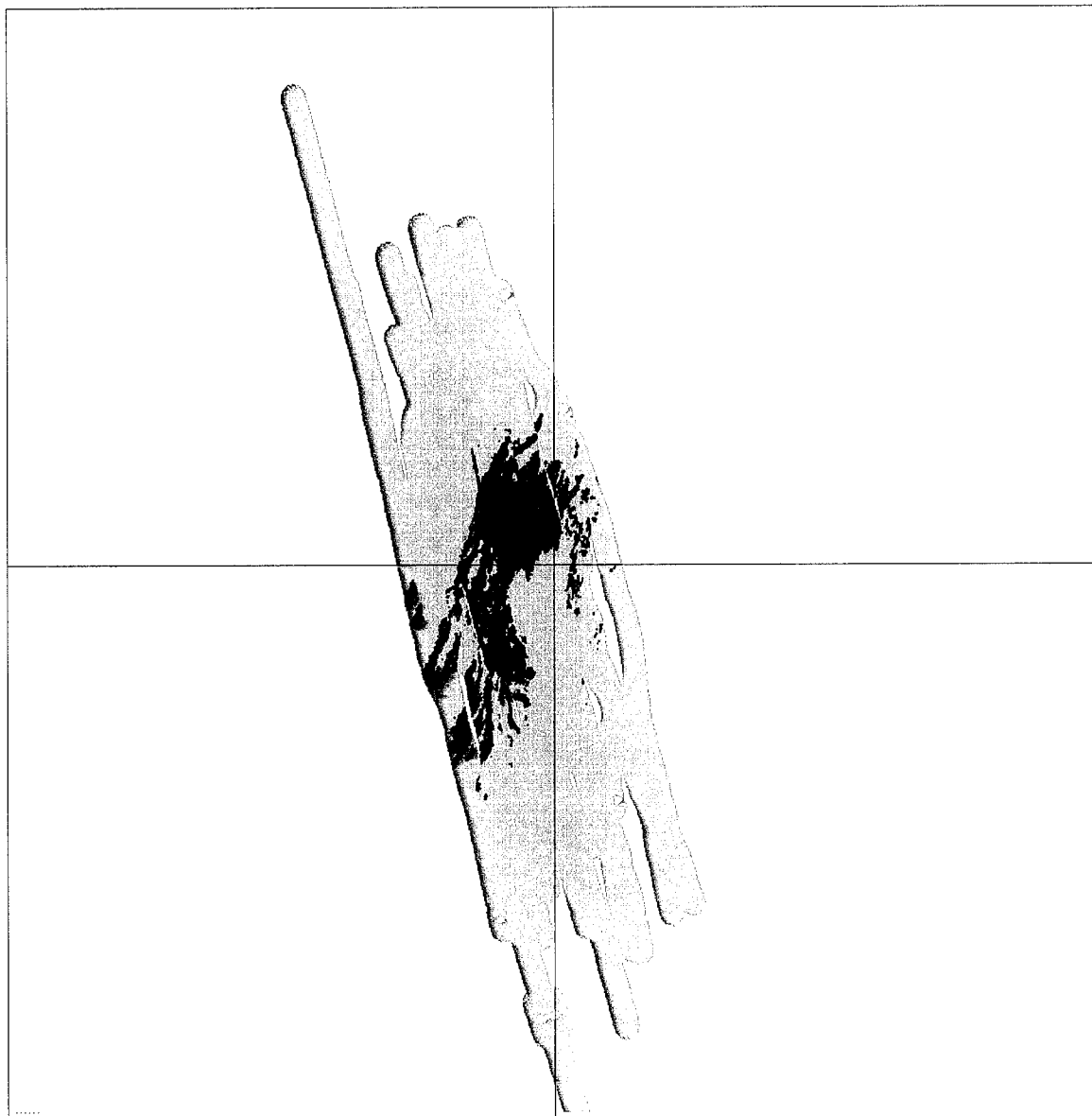
27° 30.86' N
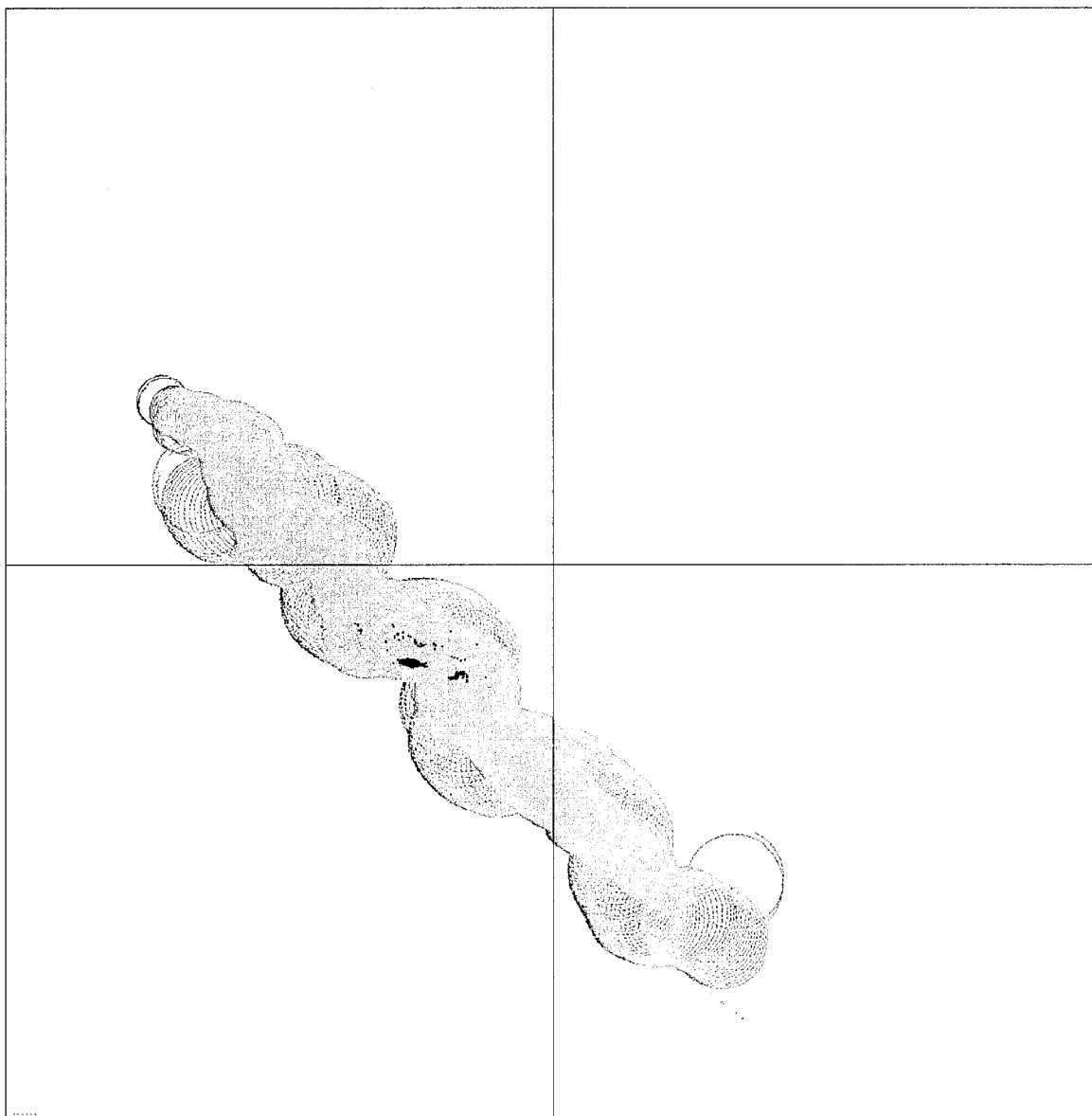
78° 52.45' W
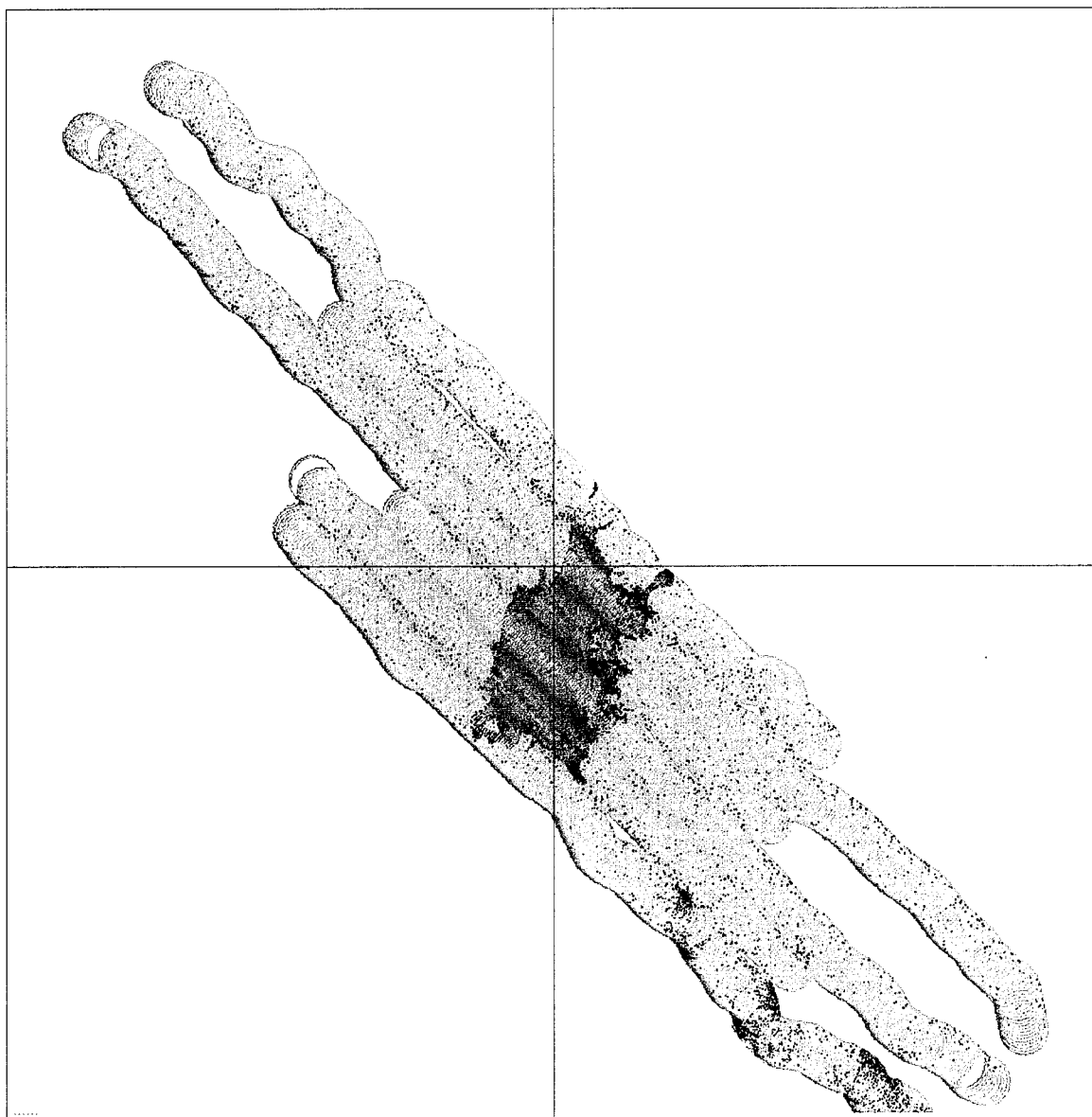
Intensity Map 19903_38.INT

(20 x 20 km)

29° 25.35' N

75° 26.71' W

Intensity Map 20006_24.INT
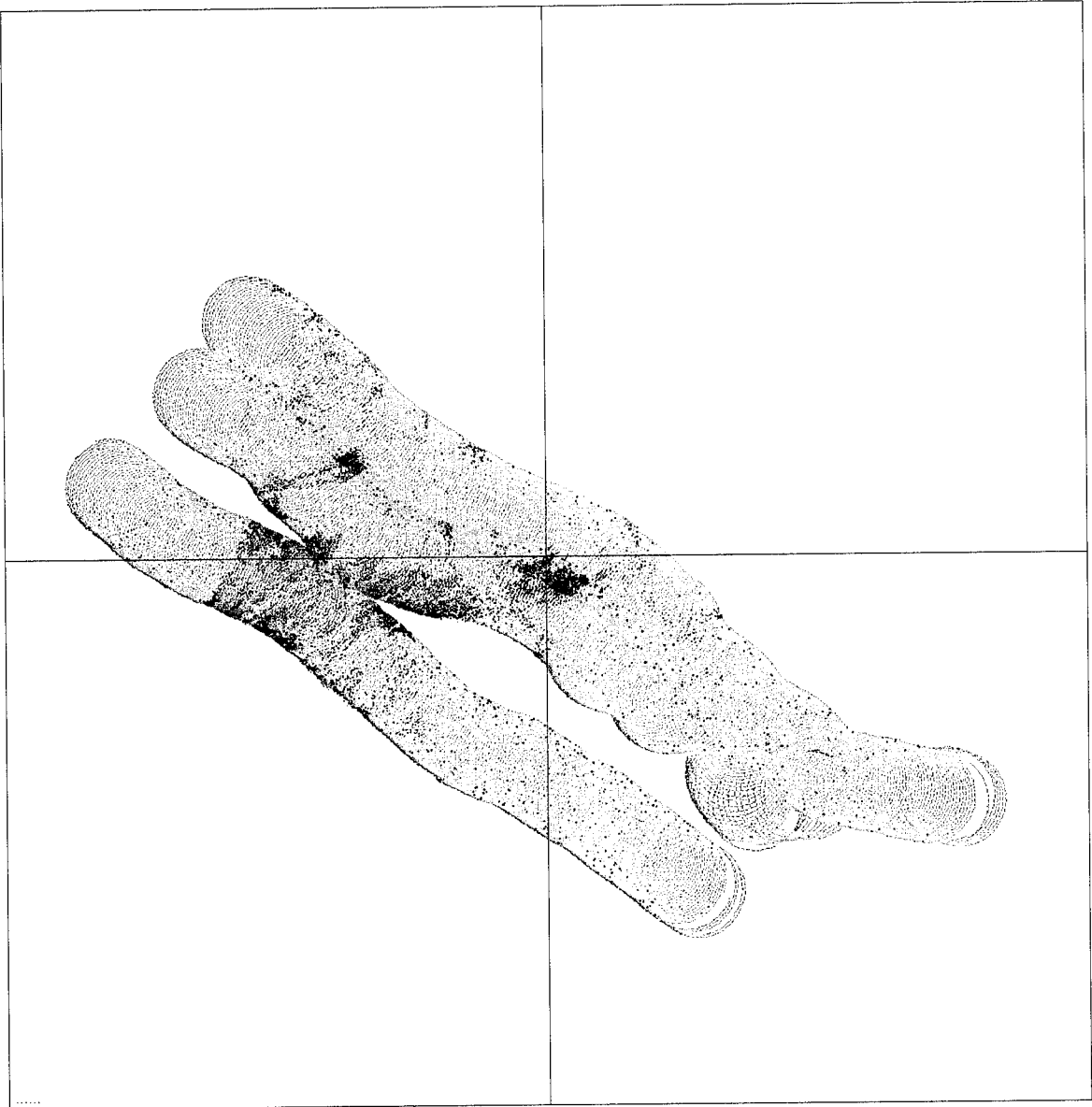
(20 x 20 km)

29° 25.04' N

75° 23.80' W

Intensity Map 20301_58.INT

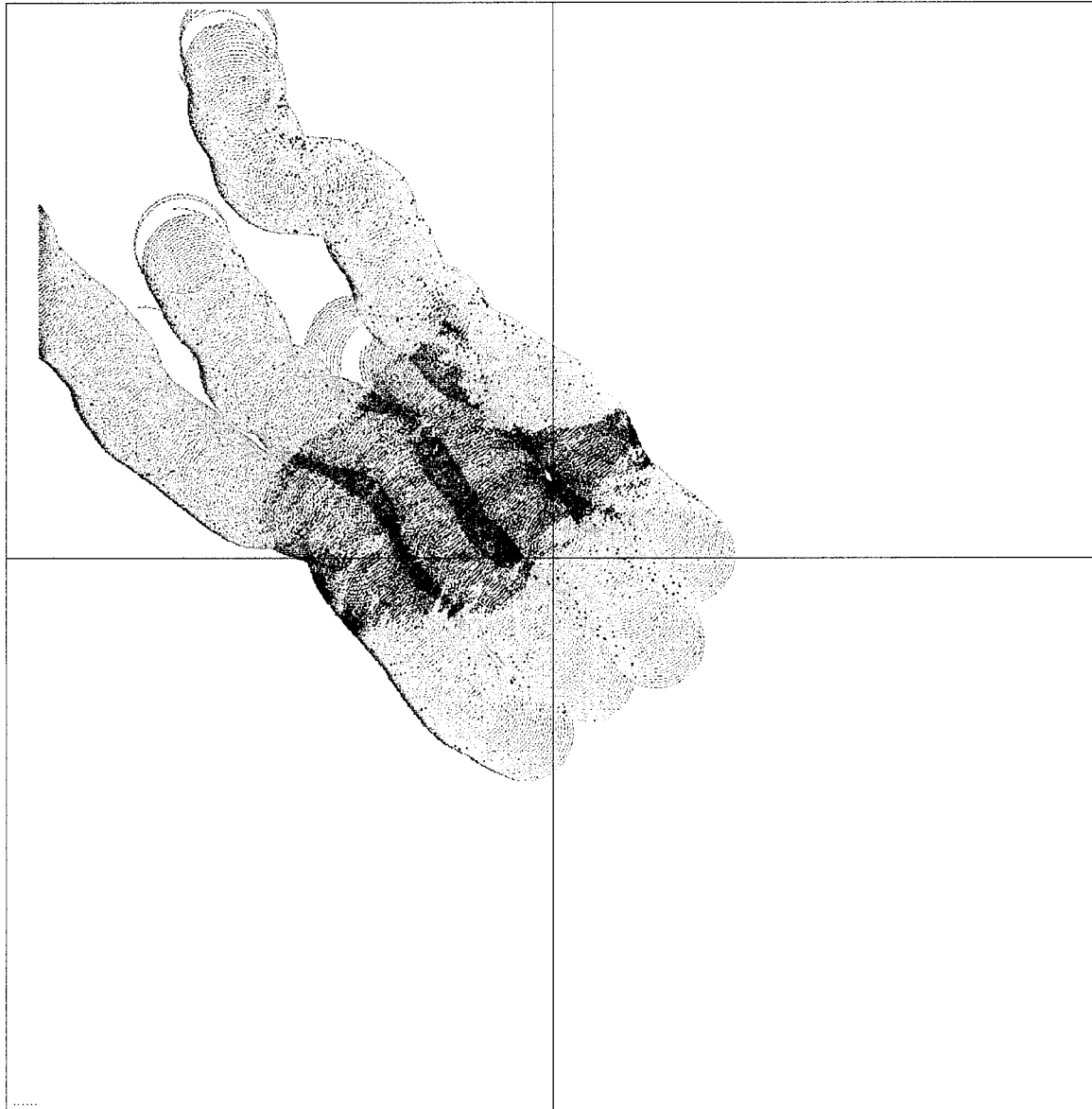(20 x 20 km)

29° 24.12' N

75° 18.27' W

Intensity Map 20402_00.INT

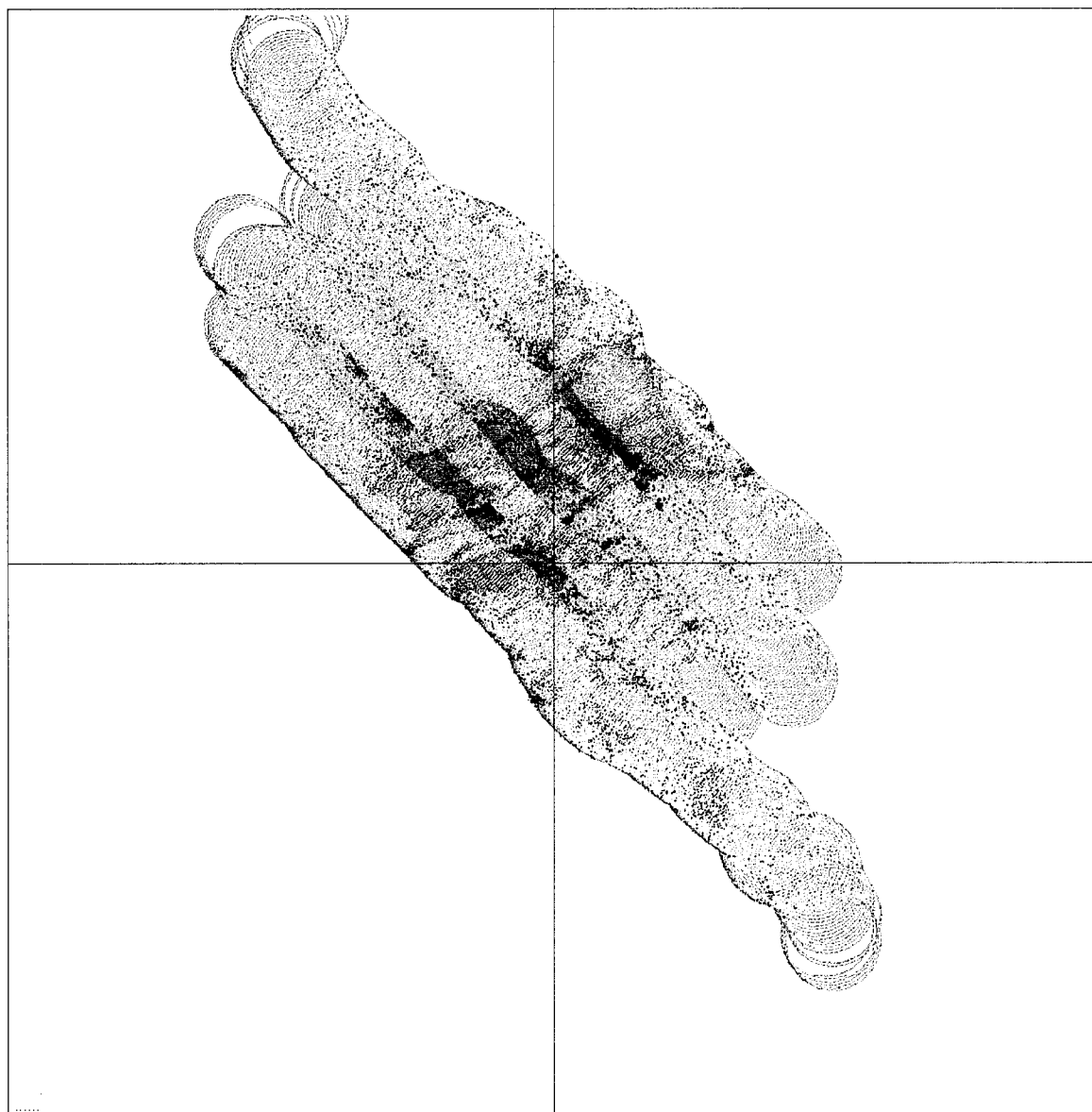(20 x 20 km)

29° 21.20' N

75° 17.91' W

Intensity Map 20801_55.INT
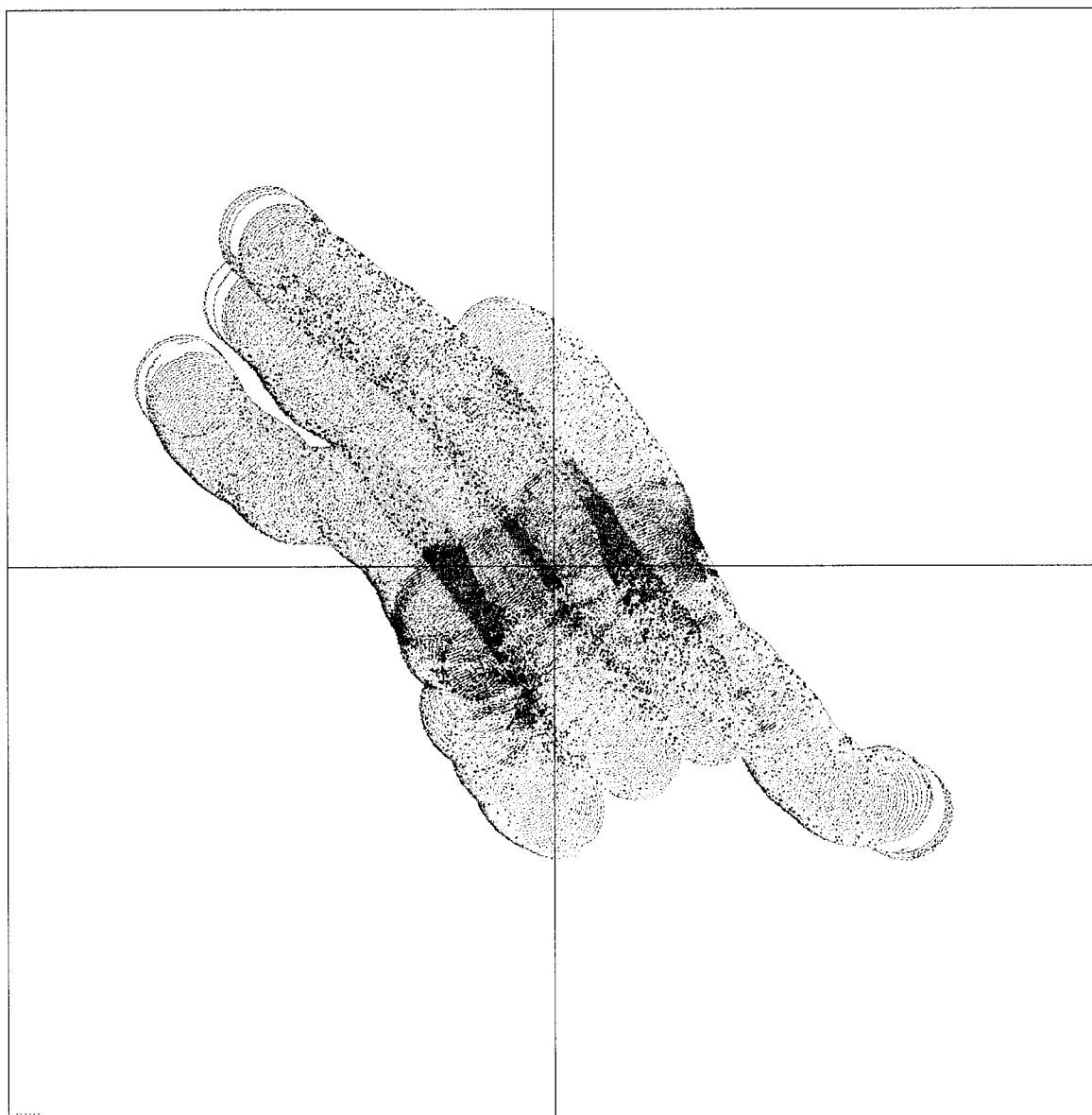
(20 x 20 km)

29° 19.21' N

75° 17.35' W

Intensity Map 20905_22.INT

(20 x 20 km)

29° 16.13' N

75° 39.96' W